



2176
AF/2176

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:

Atty. Docket:

Thrane

NC30507

Serial No.: 09/691,775

Art Unit: 2176

Filed: October 18, 2000

Examiner: Nguyen, Chau T.

Customer No.: 29683

Confirmation No.: 4383

Title: Method and Apparatus for Content Transformation for Rendering Data into a Presentation Format

APPEAL BRIEF

Commissioner for Patents
Alexandria, VA 22313-1450

Sir:

This is an Appeal Brief in regard to the final rejection of claims in the above-identified patent application. A Notice of Appeal was mailed to the USPTO on June 23, 2005. The fee under 37 C.F.R. § 41.20(b)(2) is enclosed. Please charge deposit account 50-1924 for any fee deficiency.

I. Real Party in Interest

The real party in interest is Nokia Corporation.

II. Related Appeals and Interferences

The undersigned attorney is not presently aware of any directly related appeals or interferences regarding this application.

III. Status of Claims

Claims 1 – 23 are pending in this application. Claims 1 – 23 have been rejected by the Examiner. The rejection of claims 1 – 23 is appealed.

IV. Status of Amendments

A Response was filed on April 27, 2005 after the final rejection and by an Advisory Action dated May 25, 2005 the Examiner restated the final rejection of the claims.

V. Summary of Claimed Subject Matter

Independent claim 1 is for a content transformation method (see Figs. 2, 3) operated in a client-server communication system, the method comprising: receiving a content request by a server from a client (104)(See Figs. 1, 2, 3; page 3, lines 29 – 30; page 6, lines 26 – 27); performing a first stage content transformation to generate a first stage data layout based upon said content request (222, 302, 304, 306, 308)(See Figs. 2, 3; page 5, lines 1 - 9; page 6, lines 27 – 32); performing an intermediate stage content transformation using said first stage data layout to generate an intermediate data layout (224, 310, 312, 314, 316)(See Figs. 2, 3; page 5, line 10 – page 6, line 2; page 7, lines 1 –

7); and performing a final stage content transformation using said intermediate data layout to generate a presentation format based on a device used by said client (226, 216, 320, 322, 324)(See Figs. 2, 3; page 6, lines 3 - 22; page 7, lines 8 - 12).

Dependent claim 2 is for a content transformation method otherwise as recited in claim 1, wherein performing a first stage content transformation comprises retrieving data from a database (112, 222)(See Figs. 1, 2; page 5, line 1).

Dependent claim 3 is for a content transformation method otherwise as recited in claim 2, wherein performing a first stage content transformation further comprises defining a set of first stage rules (204)(See Fig. 2; page 5, lines 1 - 9).

Dependent claim 4 is for a content transformation method otherwise as recited in claim 3, wherein performing a first stage content transformation further comprises generating the first stage data layout by transforming the data using the first stage rules (208, 114, 204, 302, 304, 306, 308)(See Figs. 2, 3; page 5, lines 1 - 9; page 6, lines 31 - 32).

Dependent claim 5 is for a content transformation method otherwise as recited in claim 2, wherein performing an intermediate stage content transformation comprises performing at least one sub-stage to generate said intermediate stage data layout (224, 208, 214)(See Figs. 2, 3, page 5, lines 10 - 12; page 7, lines 1 - 6).

Dependent claim 6 is for a content transformation method otherwise as recited in claim 5, wherein performing the at least one sub-stage comprises performing a browser-type stage using a set of browser type rules (See page 5, lines 18 – 19, 25 – 27).

Dependent claim 7 is for a content transformation method otherwise as recited in claim 5, wherein performing at least one sub-stage comprises performing an internationalization stage using a set of internationalization rules (See page 5, lines 19 – 20, 27 – 29).

Dependent claim 8 is for a content transformation method otherwise as recited in claim 5, wherein performing said at least one sub-stage comprises performing a user profile stage using a set of user profile rules (See page 5, lines 20 – 21, 31 – 32).

Dependent claim 9 is for a content transformation method otherwise as recited in claim 5, wherein performing said at least one sub-stage comprises performing an optimization stage using a set of optimization rules (See page 5, line 21, lines 32 – 33).

Dependent claim 10 is for a method otherwise as recited in claim 1, wherein performing a final stage content transformation comprises defining a set of final stage rules (226, 216)(See page 6, lines 3 – 4).

Dependent claim 11 is for a method otherwise as recited in claim 10, wherein defining a set of final stage rules comprises using said content request to define the final stage rules (See page 6, lines 9 – 14; page 7, lines 9 – 10).

Dependent claim 12 is for a method otherwise as recited in claim 1, wherein the content transformation is XSLT based content transformation using an XSLT engine (See page 4, lines 30 – 32).

Independent claim 13 is for a server (108)(See Fig. 1, page 3, line 19 – page 4, line 23) in a client-server communication system, the server comprising: a content transformation operating program (109)(See Figs. 1, 2, 3; page 3, line 30 – page 4, line 1; page 4, line 24 – page 7, line 12) line to perform operations that comprise: receiving a content request from a client (104)(See Fig. 1; page 3, lines 28 – 30); performing a first stage content transformation to generate a first stage data layout based upon said content request (222, 204, 302, 304, 306, 308)(See Figs. 2, 3; page 5, lines 1 – 9; page 6, lines 27 – 32); performing an intermediate stage content transformation using said first stage data layout to generate an intermediate data layout (224, 210, 214, 310, 312, 314, 316)(See Figs. 2, 3; page 5, line 10 – page 6, line 2; page 7, lines 1 – 7); and performing a final stage content transformation using said intermediate data layout to generate a presentation format based on a device used by said client (226, 216, 320, 322, 324)(See Figs. 2, 3; page 6, lines 3 – 22; page 7, lines 8 – 12).

Dependent claim 14 is for a server otherwise according to claim 13, wherein performing a first stage content transformation further comprises retrieving data from a database (112, 114, 222, 304)(See Figs. 1, 2; page 5, line 1).

Dependent claim 15 is for a server otherwise according to claim 14, wherein performing a first stage content transformation further comprises defining a set of first stage rules (204)(See Fig. 2; page 5, lines 1 – 9).

Dependent claim 16 is for a server otherwise according to claim 15, wherein performing a first stage content transformation further comprises generating said first stage data layout by transforming the data using the first stage rules (208, 114, 204, 302, 304, 306, 308)(See Figs. 2, 3; page 5, lines 1 – 9; page 6, lines 31 – 32).

Dependent claim 17 is for a server otherwise according to claim 13, wherein performing an intermediate stage content transformation comprises performing at least one sub-stage to generate the intermediate stage data layout (224, 208, 214, 310, 312, 314, 316)(See Figs. 2, 3; page 5, lines 10 – 12; page 7, lines 1 – 6).

Dependent claim 18 is for a server otherwise according to claim 17, wherein performing said at least one sub-stage comprises performing a browser-type stage using a set of browser-type rules (See page 5, lines 18 – 19, 25 – 27).

Dependent claim 19 is for a server otherwise according to claim 17, wherein performing said at least one sub-stage comprises performing an internationalization stage using a set of internationalization rules (See page 5, lines 19 – 20, 27 – 29).

Dependent claim 20 is for a server otherwise according to claim 17, wherein performing said at least one sub-stage comprises performing a user profile stage using a set of user profile rules (See page 5, lines 20 – 21, 31 – 32).

Dependent claim 21 is for a server otherwise according to claim 17, wherein performing said at least one sub-stage comprises performing an optimization stage using a set of optimization rules (See page 5, line 21, lines 32 – 33).

Dependent claim 22 is for a server otherwise according to claim 13, wherein performing a final stage content transformation comprises defining a set of final stage rules (226, 216)(See page 6, lines 3 – 4).

Dependent claim 23 is for a server otherwise according to claim 22, wherein defining a set of final stage rules comprises using said content request to define the final stage rules (See page 6, lines 9 – 14; page 7, lines 9 – 10).

VI. Grounds of Rejection to be Reviewed on Appeal

A. Claims 1 – 7, 9 – 19 and 21 – 23 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over United States Patent No. 6,507,857 to Yalcinalp (hereinafter “the Yalcinalp patent”) in view of United States Patent No. 6,589,291 to Boag *et al.* (hereinafter “the Boag patent”).

B. Claims 8 and 20 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over the Yalcinalp and Boag patents as applied to claims 1 – 7, 9 – 19 and 21 – 23, and further in view of United States Patent No. 6,616,700 to Thum *et al.* (hereinafter “the Thum patent”).

VII. Argument

A. Rejection of Claims 1 – 7, 9 – 19 and 21 – 23 under 35 U.S.C. § 103(a)

Claim 1

The Examiner correctly recognized that the Yalcinalp patent does not disclose performing an intermediate stage content transformation using a first stage data layout to generate an intermediate data layout. Thus, the Examiner's statement that Yalcinalp does perform a final stage content transformation "using said intermediate data layout" is not agreed with.

In fact, the technique of the Yalcinalp patent appears to rely on the use of external calls that are embedded in a style sheet as stated at, for example, Col. 6, lines 13 – 26:

"The method then proceeds to generate a transform document using the style sheet **and an external call embedded in an external component in the style sheet** (step 305). The external call is to a named component instance and may perform a function defined by the developer creating the external call. In other words, it is not required that the external call perform some function related to the transformed document or even a function related to the document processing. It may be preferable for the developer of the style sheet to insert into the external function parameters and information that may be application dependent. However, as stated above, the external call may also perform various other types of processing." (emphasis added)

When characterizing the Boag patent the Examiner states that in the instant specification the applicant "described the intermediate stage comprises a browser-type substage based

on the type of browser and internationalization stage which is specific language used by the client". This overly simplistic characterization of the embodiments of this invention is respectfully disagreed with. What is stated instead at, for example, page 5, lines 17 – 25, is the following:

“In the preferred embodiment the intermediate stage comprises a browser-type sub-stage using a set of browser-type rules, an internationalization stage using a set of internationalization rules, a user profile stage using a set of user profile rules, and a optimization stage using optimization rules. It should be noted that the number of rules and the type of rules are dependent on the operator of the server. Furthermore, the additional set of rules may be added to the intermediate stage rules or set rules may be deleted from the preferred embodiment without departing from the invention.”

In any event, the Examiner uses the Boag patent for purportedly teaching the selecting of style sheets based on variable factors, such as target device and browser, or the selected style sheets may create output in a language appropriate for the wireless connection and the target device. The Examiner continues by stating that since Boag discloses a method “for dynamically determining the most appropriate location for applying style sheets on a client request depends on the capabilities of the client device, which is similar to processing a user request document to a transformed document and formatting the transformed document specific to the client specification of Yalcinalp” that it would have been obvious to combine the teachings of the Boag and Yalcinalp patents to include

“performing an intermediate stage content transforming using the first stage data layout to generate a intermediate data layout to provide a

technique for increasing the applicability of style sheets when a style sheet tailored to a particular target environment is not readily available.”

The Examiner’s rationale for rejecting the claims is disagreed with, and is respectfully traversed.

It is pointed out that neither Yalcinalp nor the Boag patent make any suggestion of:

“performing an intermediate stage content transformation using said first stage data layout to generate an intermediate data layout; and performing a final stage content transformation using said intermediate data layout to generate a presentation format based on a device used by said client”,

as is recited in claim 1.

As was noted above, the Yalcinalp patent discloses generating “a transform document using the style sheet and an external call embedded in an external component in the style sheet”. The Boag patent discloses in FIG. 3 and at Col. 9, line 46 to col. 10, line 41, the following technique:

“If the test at Block 305 has a negative result, then control transfers to Block 310; otherwise, processing continues at Block 325.

At Block 310, a style sheet tailored to the target environment is not available. **The style sheet which has been selected may therefore be a more general-purpose style sheet which is capable of performing some**

type of desired filtering of data. For example, application of a content reducing style sheet which maintains the original document type might be beneficial. As another example, transforms that manipulate data for presentation without reference to a particular presentation medium (such as selecting out, or hiding, data matching specified patterns) may be beneficial. This approach enables the applicability of style sheets to be increased, taking advantage of the transformations coded therein whenever a benefit can be realized. Block 310 will apply the selected style sheets at the server. **The resulting document may then be forwarded to a general purpose transcoding engine, as shown at Block 315. This is an optional step where more general purpose transformations (such as converting from one markup language to another) may be applied, and uses techniques which do not form part of the present invention. As one example, a general purpose transcoder which converts HTML to WML may be used,** such as the Prism transcoder, which is commercially available from Spyglass, Inc. As another example, a transcoding algorithm may be applied to perform 256-color to 16-color reduction for all image files being sent over wireless connections. As yet another example, the novel techniques disclosed in U.S. Pat. No. 6,138,156, which is titled "Selecting and Applying Content-Reducing Filters based on Dynamic Environmental Factors" and is assigned to the same assignee, may be used. This invention describes a technique whereby environmental factors can be used to dynamically filter the content being delivered from a server to a user's workstation. **The choice of an appropriate general purpose transcoder is preferably determined using characteristics of the target device and browser.** This information may be obtained, for example, by inspecting the UserAgent field of the HTTP request. (In addition, protocols are under development for querying a device to determine this type of information, and could be used for this purpose. One such approach is the 'Composite

Capability/Preference Profiles’, or ‘CC/PP’, under development by W3C.)
If no general purpose transcoding algorithm is located which is suitable for this document, Block 315 may be skipped, or an exception generated.

By invoking a general purpose transcoding engine, the present invention may be used advantageously with a new device type for which no specific style sheets are yet available. Suppose the new device requires documents encoded in WML, and the style sheet selection process of Block 300 determined that (in the absence of a style sheet tailored to this device) the best choice was a content-reducing style sheet which produces HTML. **Block 315 may then invoke a transcoder capable of converting HTML to WML**, as described above. In this manner, the benefit of the general content-reducing filter is used, while still being able to generate content in markup language which the device can process.”
(emphasis added)

There is thus clearly no suggestion in the Boag patent of:

“performing an intermediate stage content transformation using said first stage data layout to generate an intermediate data layout; and
performing a final stage content transformation using said intermediate data layout to generate a presentation format based on a device used by said client”,

as is recited in claim 1. **In fact, the invocation of a particular transcoder (e.g., a HTML to WML transcoder) to operate on the selected style sheet appears to instead be a function that one could implement by using the embedded external calls of Yalcinalp** (see, for example, Yalcinalp at Col. 6, lines 13 – 26, and col. 9, lines 31 – 38).

That is, if one did attempt to combine the teachings of the Yalcinalp and Boag patents in the manner attempted by the Examiner, which is not admitted is suggested or technically feasible, the resulting hybrid system would appear at most to simply use the embedded external call of Yalcinalp to invoke the transcoder of Boag. What would clearly **not be suggested** to one of ordinary skill in the art would be performing **an intermediate stage content transformation using a first stage data layout to generate an intermediate data layout; and performing a final stage content transformation using the intermediate data layout to generate a presentation format based on a device used by a client.**

Applicant therefore respectfully submits that claim 1 is patentable and should be allowed.

Claims 2 – 7 and 9 – 12 stand or fall with claim 1.

Claim 13

Applicant respectfully notes that claim 13 recites the following subject matter:

“performing an intermediate stage content transformation using said first stage data layout to generate an intermediate data layout; and performing a final stage content transformation using said intermediate data layout to generate a presentation format based on a device used by said client.”

Accordingly, all the foregoing arguments presented with respect to claim 1 are similarly applicable to claim 13.

Additionally, Applicant observes that claim 13 is directed to a *server*, and all the content transformation steps recited in claim 13 are performed by the *server*. It is not seen where the references of record either describe or suggest the performance of first stage, intermediate stage, and final stage content transformation steps at a server which has received a content request from a client.

First, as stated previously, the Examiner has reiterated in the most recent Office Action that “Yalcinalp does not explicitly disclose performing an intermediate state content transforming using said first stage data layout to generate a intermediate data layout.” (February 4, 2005 Office Action, page 3, lines 8 – 10) Second, and in particular, the Boag patent does not concern methods and apparatus for performing content transformation steps, including an intermediate stage content transformation, solely at a server in a client-server system. Rather, as shown by Col. 4, lines 23 – 49, the Boag patent concerns a method and system that chooses the appropriate location to apply style sheets depending on the capability of client and server devices:

“To achieve the foregoing objects, and in accordance with the purpose of the invention as broadly described herein, the present invention provides a method, system, and computer-readable code for use in a computing environment capable of having a connection to a network, **for dynamically determining the most appropriate location to apply style sheets. The technique comprises: selecting one or more style sheets to**

transform a particular input document; determining whether a client device is capable of applying the selected style sheets; applying the selected style sheets at the client device when the determining has a positive result; and applying the selected style sheets at a server when the determining has a negative result. The determining may determine a first subset of the selected style sheets that the client device is capable of applying, and a second subset that the client device is not capable of applying. In that case, the applying at the client step applies the first subset, and the applying at the server step applies the to [sic] second subset.

In one aspect, when the determining has a positive result, the technique may further comprise: caching the selected style sheets at the server; modifying the input document to refer to the cached style sheets; and sending the modified input document to the client. When the determining has a negative result, the technique may further comprise: sending the input document to the client following completion of the applying at the server.” (emphasis added)

Applicant respectfully observes that his invention does not operate in this manner; the content transformation steps are performed at the server.

For example, the operation of Applicant's invention is described here:

“In an exemplary implementation of the client-server communication system 100 the client 101 is typically connected to the server 108 via client-server link 102. The client-server link 102 may comprise a wireless link or an electronic link, such as a telephone connection. The client 101 comprises a software program, such as a browser, to allow the user to create and send a content request 104. The client 101 may be a mobile terminal, general purpose computer, a Personal

Digital Assistant (PDA) or other client terminal device having the browser. The server 108 may be a general-purpose computer having a memory and processor. The server 108 may be connected to one or more clients analogous to client 101. The server 108 comprises plurality of operating programs which receive user's search request, such as the content request 104. **The server 108 further comprises multi-stage content transformation operating program 109 for carrying out an embodiment of the invention.** The server 108 is typically connected to the database 112 via a server-database link 110 for retrieving data based on the content request 104 wherein the server-database link 110 may be a wireless or an electronic link." [Application, Page 3, line 19 – page 4, line 4] (emphasis added)

* * *

"FIG. 2 illustrates a preferred embodiment of a multi-stage content transformation process 200. The content is transformed in three stages, a first stage content transformation 222, an intermediate stage content transformation 224 and a final stage content transformation 226. Each stage comprises one or more stage rules (also known as formatting template or stylesheet) defining the layout of a Meta Markup Language (MML) document wherein data is embedded according to rules. In the preferred embodiment, a generic Extensive Stylesheet Language Transformation (XSLT) engine is used to merge (transform) data according to the rules in each stage." [Application, Page 4, lines 24 - 32] (emphasis added)

* * *

"The intermediate stage 224 comprises one or more sub-stages that are executed using one or more intermediate rules to further transform first stage data layout 208 to intermediate stage data layout 214 . . . When all the rules are merged (or all the sub-stages are executed), the intermediate stage data layout is created. In the

preferred embodiment the intermediate stage comprises a browser-type sub-stage using a set of browser-type rules, an internationalization stage using a set of internationalization rules, a user profile stage using a set of user profile rules, and a optimization stage using optimization rules.” [Application, Page 5, lines 20 – 21]
(emphasis added)

It is not seen where the Boag patent, or any of the other references, either describe or suggest a content transformation process performed at a server having an intermediate content transformation stage operating in the manner of Applicant’s invention. If the Examiner disagrees, Applicant respectfully requests that the Examiner identify with particularity exactly where such subject matter is disclosed.

In addition, when performing a complex set of transformations, the Boag patent teaches away from performing a distinct intermediate stage content transformation as in the case of Applicant’s invention as shown at Col. 11, lines 11 - 16:

“(When a sequence of style sheets is to be applied to perform a complex transformation, the technique disclosed in the first related invention will preferably be used to chain the multiple style sheets together. The modified reference in the input document is then a reference to where the chained style sheets are cached.)”

Notably, neither in this portion, nor in any other portion of the Boag patent, is there a description or suggestion of performing at a server an intermediate stage content transformation followed by a final stage content transformation.

Applicant therefore respectfully submits that claim 13 is patentable and should be allowed.

Claims 14 – 19 and 21 - 23 stand or fall with claim 13.

**B. Rejection of Claims 8 and 20 under
35 U.S.C. § 103(a)**

Applicant respectfully submits that claims 8 and 20 are patentable as being indirectly dependent on base claims (independent claims 1 and 13) that are patentable for the foregoing reasons. Additionally, Applicant respectfully submits that the disclosure of the Thum does not cure the deficiencies of either the Yalcinalp or Boag patents.

Applicant therefore respectfully submits that claims 8 and 20 are patentable and should be allowed.

Conclusion

In view of the arguments presented above, it is respectfully requested that the Examiner's rejection of claims 1 – 23 be reversed.

Respectfully submitted,

August 23, 2005

Date

David M. O'Neill (35,304)

David M. O'Neill (35,304)

Customer No.: 29683

HARRINGTON & SMITH, LLP
4 Research Drive
Shelton, CT 06484-6212
Telephone: (203) 925-9400
Facsimile: (203) 944-0245
Email: DOneill@hspatent.com

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Commissioner for Patents, P.O. 1450, Alexandria, VA 22313-1450 on the date indicated.

8/23/05

Date

Ann O'Brien Towich

Name of Person Making Deposit



VIII. CLAIMS APPENDIX



A content transformation method operated in a client-server communication system, the method comprising:

receiving a content request by a server from a client;

performing a first stage content transformation to generate a first stage data layout based upon said content request;

performing an intermediate stage content transformation using said first stage data layout to generate an intermediate data layout; and

performing a final stage content transformation using said intermediate data layout to generate a presentation format based on a device used by said client.

2. The method as recited in claim 1, wherein performing a first stage content transformation comprises retrieving data from a database.

3. The method as recited in claim 2, wherein performing a first stage content transformation further comprises defining a set of first stage rules.

4. The method as recited in claim 3, wherein performing a first stage content transformation further comprises generating said first stage data layout by transforming said data using said first stage rules.

5. The method as recited in claim 2, wherein performing an intermediate stage content transformation comprises performing at least one sub-stage to generate said intermediate stage data layout.

6. The method as recited in claim 5, wherein performing said at least one sub-stage comprises performing a browser-type stage using a set of browser-type rules.

7. The method as recited in claim 5, wherein performing said at least one sub-stage comprises performing an internationalization stage using a set of internationalization rules.

8. The method as recited in claim 5, wherein performing said at least one sub-stage comprises performing a user profile stage using a set of user profile rules.

9. The method as recited in claim 5, wherein performing said at least one sub-stage comprises performing an optimization stage using a set of optimization rules.

10. The method as recited in claim 1, wherein performing a final stage content transformation comprises defining a set of final stage rules.

11. The method as recited in claim 10, wherein defining a set of final stage rules comprises using said content request to define the final stage rules.

12. The method as recited in claim 1, wherein the content transformation is XSLT based content transformation using an XSLT engine.

13. A server in a client-server communication system, the server comprising;

a content transformation operating program to perform operations that comprise:

- receiving a content request from a client;
- performing a first stage content transformation to generate a first stage data layout based upon said content request;
- performing an intermediate stage content transformation using said first stage data layout to generate an intermediate data layout; and
- performing a final stage content transformation using said intermediate data layout to generate a presentation format based on a device used by said client.

14. The server according to claim 13, wherein performing a first stage content transformation, further comprises retrieving data from a database.

15. The server according to claim 14, wherein performing a first stage content transformation, further comprises defining a set of first stage rules.

16. The server according to claim 15, wherein performing a first stage content transformation further comprises generating said first stage data layout by transforming said data using said first stage rules.

17. The server according to claim 13, wherein performing an intermediate stage content transformation comprises performing at least one sub-stage to generate said intermediate stage data layout.

18. The server according to claim 17, wherein performing said at least one sub-stage comprises performing a browser-type stage using a set of browser-type rules.

19. The server according to claim 17, wherein performing said at least one sub-stage comprises performing an internationalization stage using a set of internationalization rules.

20. The server according to claim 17, wherein performing said at least one sub-stage comprises performing a user profile stage using a set of user profile rules.

21. The server according to claim 17, wherein performing said at least one sub-stage comprises performing an optimization stage using a set of optimization rules.

22. The server according to claim 13, wherein performing a final stage content transformation comprises defining a set of final stage rules.

23. The server according to claim 22, wherein defining a set of final stage rules comprises using said content request to define the final stage rules.